PORTAL USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

Searching within **The ACM Digital Library** with **Advanced Search**: (Title:intermediate and Title:language) (start a new search)

Found **20** of **252,873**

REFINE YOUR SEARCH Search Results Related SIGs Related Conferences

Refine by Keywords

Discovered Terms

Refine by People
Names
Institutions
Authors

Refine by Publications
Publication Year
Publication Names
ACM Publications
All Publications
Content Formats
Publishers

Refine by Conferences
Sponsors
Events
Proceeding Series

ADVANCED SEARCH
Advanced Search

FEEDBACK
Please provide us with feedback

Found **20** of **252,873**

Results 1 - 20 of 20 Sort by relevance in expanded form

Save results to a Binder

1 Transforming the .NET intermediate language using path logic programming

Stephen Drape, Oege de Moor, Ganesh Sittampalam

October 2002 **PPDP '02:** Proceedings of the 4th ACM SIGPLAN international conference o Principles and practice of declarative programming

Publisher: ACM Request Permissions

Full text available: Pdf (251.91 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 0, Downloads (12 Months): 23, Citation Count: 4

Path logic programming is a modest extension of Prolog for the specification of progr transformations. We give an informal introduction to this extension, and we show ho can be used in coding standard compiler optimisations, and also a number of ...

Keywords: compiler optimisations, logic programming, meta programming, obfuscation, program analysis, program transformation

2 P-TAC: a parallel intermediate language

Zena Ariola, Arvind

November 1989 **FPCA '89:** Proceedings of the fourth international conference on Functio programming languages and computer architecture

Publisher: ACM Request Permissions

Full text available: Pdf (1.47 MB) Additional Information: full citation, references, cited by, index terms

Bibliometrics: Downloads (6 Weeks): 2, Downloads (12 Months): 14, Citation Count: 6

3 Representing Java classes in a typed intermediate language

Christopher League, Zhong Shao, Valery Trifonov

September 1999 **ICFP '99:** Proceedings of the fourth ACM SIGPLAN international confere on Functional programming

Publisher: ACM Request Permissions

Full text available: Pdf (1.81 MB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 1, Downloads (12 Months): 18, Citation Count: 12

We propose a conservative extension of the polymorphic lambda calculus ($F^{\ω}$) as intermediate language for compiling languages with name-based class and interface hierarchies. Our extension enriches standard $F^{\ω}$...

Also published in:
September 1999 **SIGPLAN Notices** Volume 34 Issue 9

4 The Mahler experience: using an intermediate language as the machine descript

David W. Wall, Michael L. Powell

November 1987 **ASPLOS-II:** Proceedings of the second international conference on

Architectual support for programming languages and operating systems

Publisher: ACM

Full text available: Pdf (618.75 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 5, Downloads (12 Months): 24, Citation Count: 14

Division of a compiler into a front end and a back end that communicate via an intermediate language is a well-known technique. We go farther and use the intermediate language as the official description of a family of machines with simple instruction ...

Also published in:

October 1987	**SIGPLAN Notices**	Volume 22 Issue 10
October 1987	**SIGOPS Operating Systems Review**	Volume 21 Issue 4
November 1987	**SIGARCH Computer Architecture News**	Volume 15 Issue 5

5 Direct microprogrammed execution of the intermediate text from a high-level language compiler

Francois Robert Broca, Richard E. Merwin

August 1973 **ACM'73:** Proceedings of the annual conference

Publisher: ACM Request Permissions

Full text available: Pdf (521.76 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 0, Downloads (12 Months): 2, Citation Count: 3

Microprogramming commonly executed operations can improve the computational sp of data processing systems. This paper describes how microprogramming may be use to execute directly the intermediate text generated by a high-level language compile

6 An intermediate language for source and target independent code optimization

Dennis J. Frailey

August 1979 **SIGPLAN '79:** Proceedings of the 1979 SIGPLAN symposium on Compiler construction

Publisher: ACM

Full text available: Pdf (864.12 KB) Additional Information: full citation, abstract, references, index terms

Bibliometrics: Downloads (6 Weeks): 3, Downloads (12 Months): 20, Citation Count: 0

This paper describes an intermediate language to be generated by a syntax analyzer processed by a code generator. An (essentially) optional code optimization phase ma be used before code generation. The language is designed to exclude source and ...

Also published in:

August 1979 **SIGPLAN Notices** Volume 14 Issue 8

7 A single intermediate language that supports multiple implementations of exceptions

Norman Ramsey, Simon Peyton Jones

August 2000 **PLDI '00:** Proceedings of the ACM SIGPLAN 2000 conference on Programm language design and implementation

Publisher: ACM Request Permissions

Full text available: Pdf (900.75 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 5, Downloads (12 Months): 49, Citation Count: 12

We present mechanisms that enable our compiler-target language, C--, to express fo of the best known techniques for implementing exceptions, all within a single, uniforr

framework. We define the mechanisms precisely, using a formal operational semanti ...

Also published in:
May 2000 **SIGPLAN Notices** Volume 35 Issue 5

8 An intermediate language for vector processors
Abbas S. Youssefi, D. E. Stevenson
April 1990 **ACM-SE 28:** Proceedings of the 28th annual Southeast regional conference
Publisher: ACM Request Permissions
Additional Information: full citation, index terms

Bibliometrics: Downloads (6 Weeks): n/a, Downloads (12 Months): n/a, Citation Count: 0

9 A comparison of PASCAL intermediate languages
Philip A. Nelson
August 1979 **SIGPLAN '79:** Proceedings of the 1979 SIGPLAN symposium on Compiler construction
Publisher: ACM
Full text available: Pdf (380.92 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 4, Downloads (12 Months): 20, Citation Count: 2

PASCAL P-code is an aid when transporting PASCAL to new machines and environmei Implementations using P-code may modify the P-code. This modification process may require some creative effort. Some of this effort may be eliminated by looking at previous ...

Also published in:
August 1979 **SIGPLAN Notices** Volume 14 Issue 8

10 A functional intermediate form for diverse source languages
David V. Mason
November 1996 **CASCON '96:** Proceedings of the 1996 conference of the Centre for Advanced Studies on Collaborative research
Publisher: IBM Press
Full text available: Pdf (215.24 KB) Additional Information: full citation, abstract, references, index terms

Bibliometrics: Downloads (6 Weeks): 0, Downloads (12 Months): 7, Citation Count: 0

This paper describes the use of WatIF, a simple compiler intermediate form based on λ-calculus, for writing compiler front-ends for a wide range of programming language Somewhat similar intermediate forms have been previously used for single-language

11 Implementing typed intermediate languages
Zhong Shao, Christopher League, Stefan Monnier
January 1999 **ICFP '98:** Proceedings of the third ACM SIGPLAN international conference Functional programming
Publisher: ACM Request Permissions
Full text available: Pdf (1.50 MB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 2, Downloads (12 Months): 17, Citation Count: 16

Recent advances in compiler technology have demonstrated the benefits of using strongly typed intermediate languages to compile richly typed source languages (e.g. ML). A type-preserving compiler can use types to guide advanced optimizations and to ...

Also published in:

January 1999 **SIGPLAN Notices** Volume 34 Issue 1

12 Refined types: highly differentiated type systems and their use in the design of intermediate languages

J. R. Rose

July 1988 **PLDI '88:** Proceedings of the ACM SIGPLAN 1988 conference on Programming Language design and Implementation

Publisher: ACM Request Permissions

Full text available: Pdf (1.23 MB) Additional Information: full citation, references, index terms

Bibliometrics: Downloads (6 Weeks): 2, Downloads (12 Months): 17, Citation Count: 0

Also published in:

July 1988 **SIGPLAN Notices** Volume 23 Issue 7

13 An intermediate design language and its analysis

Daniel Jackson

November 1998 **SIGSOFT '98/ FSE-6:** Proceedings of the 6th ACM SIGSOFT international symposium on Foundations of software engineering

Publisher: ACM Request Permissions

Full text available: Pdf (901.37 KB) Additional Information: full citation, abstract, references, cited by, index terms

Bibliometrics: Downloads (6 Weeks): 2, Downloads (12 Months): 13, Citation Count: 5

A simple relational language is presented that has two desirable properties. First, it is sufficiently expressive to encode, fairly naturally, a variety of software design problem Second, it is amenable to fully automatic analysis. This paper explains ...

Also published in:

November 1998 **SIGSOFT Software Engineering Notes** Volume 23 Issue 6

14 An intermediate systems language for the PDP-11

Peter Karasz

July 1977 **SIGPLAN Notices**, Volume 12 Issue 7

Publisher: ACM

Full text available: Pdf (1.23 MB) Additional Information: full citation, abstract

Bibliometrics: Downloads (6 Weeks): 1, Downloads (12 Months): 4, Citation Count: 0

In this paper we describe Version 2 of Intermediate Systems Language (ISL/V2), a general purpose software implementation language developed at the B. F. Goodrich Company. The particular implementation we discuss here is for the PDP-11 family of minicomputers ...

15 An aspect-oriented infrastructure for a typed, stack-based, intermediate assembly language

Douglas R. Dechow

November 2002 **OOPSLA '02:** Companion of the 17th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications

Publisher: ACM Request Permissions

Full text available: Pdf (162.00 KB) Additional Information: full citation, abstract, references

Bibliometrics: Downloads (6 Weeks): 0, Downloads (12 Months): 7, Citation Count: 0

While traditional, one-dimensional approaches to the problem of separation of concer have been adequate for current software development, they are often brittle and resistant to evolutionary change. Aspects and aspect-orientation offer a controllable,

Keywords: aspect weaver, aspect-oriented programming, aspects, common intermediate language (CIL), compile-time, components, ecological modeling, preprocessors, runtime, separation of concerns, weave-time

16 Typing a multi-language intermediate code

Andrew D. Gordon, Don Syme

January 2001 **POPL '01:** Proceedings of the 28th ACM SIGPLAN-SIGACT symposium on Principles of programming languages

Publisher: ACM Request Permissions

Full text available: Pdf (257.05 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 8, Downloads (12 Months): 66, Citation Count: 19

The Microsoft .NET Framework is a new computing architecture designed to support a variety of distributed applications and web-based services. .NET software component are typically distributed in an object-oriented intermediate language, Microsoft IL, ...

Also published in:

March 2001 **SIGPLAN Notices** Volume 36 Issue 3

17 The direct execution of intermediate languages on an Eclipse computer

R. E. M. Cooper

March 1980 **SIGMICRO Newsletter** , Volume 11 Issue 1

Publisher: ACM

Full text available: Pdf (332.48 KB) Additional Information: full citation, abstract, references

Bibliometrics: Downloads (6 Weeks): 0, Downloads (12 Months): 8, Citation Count: 0

The use of intermediate languages as a convenient means of developing portable hig level language compilers, appears to be growing. With this approach, the compiler fo language A (which may be written in language A) compiles the source code into an ..

18 FLIC—a functional language intermediate code

S. L. Payton Jones

August 1988 **SIGPLAN Notices** , Volume 23 Issue 8

Publisher: ACM

Full text available: Pdf (1.19 MB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 0, Downloads (12 Months): 5, Citation Count: 3

FLIC is a Functional Language Intermediate Code, intended to provide a common intermediate language between diverse implementations of functional languages, including parallel ones. This paper gives a formal definition of FLIC's syntax and semantics, ...

19 Intermediate language for minicomputer cross-compilation

Joseph K. Speetjens

April 1976 **ACM-SE 14:** Proceedings of the 14th annual Southeast regional conference

Publisher: ACM Request Permissions

Full text available: Pdf (266.11 KB) Additional Information: full citation, abstract, references

Bibliometrics: Downloads (6 Weeks): 2, Downloads (12 Months): 6, Citation Count: 0

With the increasing use of minicomputers, the concept of cross-compilation (source program translated by one machine which in turn outputs object code for another machine) has become popular. This paper examines the use of an intermediate language, a ...

20

Direct microprogrammed execution of the intermediate text from a high-level language compiler

Francois Robert Broca, Richard E. Merwin

August 1974 Proceedings of the meeting on SIGPLAN/SIGMICRO interface
Publisher: ACM

Full text available: Pdf (654.98 KB) Additional Information: full citation, abstract, references, cited by, inc terms

Bibliometrics: Downloads (6 Weeks): 1, Downloads (12 Months): 9, Citation Count: 8

Microprogramming commonly executed operations can improve the computational sp of data processing systems. This paper describes how microprogramming may be use to execute directly the intermediate text generated by a high-level language compile

Also published in:
August 1974 **SIGPLAN Notices** Volume 9 Issue 8

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2009 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player